An Interop Publication

ConneXions®

The Interoperability Report

September 1996 Volume 10, No. 9

ConneXions —
The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.

In this issue:

ConneXions is published monthly by Interop Company, a division of SOFTBANK Exposition and Conference Company, 303 Vintage Park Drive, Foster City, California, 94404–1138, USA.
Phone: +1 (415) 578-6900
Fax: +1 (415) 525-0194
E-mail: connexions@interop.com

Subscription hotline: 1-800-575-5717
or +1 610-892-1959

ISSN 0894–5926

From the Editor

As outlined in our May issue, Version 2 of the *Simple Network Management Protocol* (SNMP) is "in limbo" awaiting the resolution of issues relating to security. An advisory team was recently formed to analyze the two existing SNMP security specifications (USEC and SNMPv2*). The purpose of the *Security and Administrative Framework Evolution for SNMP Advisory Team* is to provide input to the network management community. By November 1, they will produce a white paper that identifies similarities and differences between the two proposals. For the areas of differences, the team will attempt to recommend a solution. The intention is to activate a working group at the next Internet Engineering Task Force (IETF) meeting in December 1996, (San Jose, CA) that will create a specification for SNMP security. Details of the team's plan can be found in a recent posting to the IETF mailing list. To join this mailing list, send a message to: `ietf-announce-request@ietf.org.`

Meanwhile, Version 1 of SNMP has seen widespread deployment in all areas of computer communications. The introduction of HTTP and Web browsers has led many to propose the use of these new technologies for network monitoring and control. Our first article, by Chris Wellens and Karl Auerbach, discusses the marriage of SNMP with HTTP, and considers new concepts and capabilities in network management.

Debate over the so-called *Clipper Chip* for encrypted communication has been widespread for a couple of years. Charlie Kaufman, Radia Perlman and Mike Speciner give an overview of the Clipper technology as well as the controversy surrounding it. The text is adapted from their excellent book *Network Security: Private Communication in a Public World.* We hope to have a review of this book in an upcoming issue.

In June we published an opinion piece on the *Communications Decency Act* (CDA). Shortly after our publication, a panel of three judges of the U.S. District Court for the Eastern District of Pennsylvania decided for the plaintiffs against the federal government over the CDA. We asked John Quarterman for an analysis of their findings.

Interop Online is our Web site, at `http://www.interop.com.` Anne Ryder gives an overview of this new networking resource on page 26. We encourage you to visit the site and give us feedback.

As always, we are looking for comments and suggestions regarding anything you read in *ConneXions.* The best way to provide this input is to send e-mail to: `connexions@interop.com.`

Towards Useful Management

by Chris Wellens, Interworking Labs
and
Karl Auerbach, Precept Software

Introduction

The *Simple Network Management Protocol* (SNMP)'s greatest success is in providing the framework to deliver management capabilities for highly focused, device specific applications. The industry needs to move beyond this accomplishment.

The purpose of this article is to consider new concepts and capabilities in network management. These range from incremental enhancements of the current state of affairs to wild-eyed dreaming.

The approaches are these, in order of increasing departure from current practices:

- Enhanced *Management Information Base* (MIB) definitions with greatly increased MIB semantics, in particular, the creation of "meta variables";
- Embedding of management applications into devices, with control interfaces exported to humans via HTTP/HTML based Web pages;
- Replacement of the SNMP access method with one based on HTTP;
- Replacement of the SNMP access method with one based on long term "associations";
- Simple management by delegation through the use of script MIBs;
- Semi-autonomous area managers; and,
- Network management "worms."

These approaches are not mutually exclusive.

MIBs are precious

During the SNMP years, we've come up with reams and reams of MIB definitions. These MIBs comprise the collected thoughts by experts of what exactly constitutes the valuable data points needed to monitor and control a device. These MIBs are the most valuable legacy of SNMP.

Along with the MIBs themselves, we have learned the value of concise, machine-parseable MIB definitions. These form the fundamental vehicle by which a general purpose management station can learn about the devices under its control.

Myths

Network management in the Internet has been the product of many myths. At least two of these have been shown to be mere vapors:

The Myth of The Collapsing Network

Connectionless transports, such as UDP, have been advanced as necessary for network management because of their ability to work when the network is failing. To put the myth contrariwise, management using TCP is deemed impossible because the myth asserts that TCP streams will break but that trusty UDP will get through to save the day.

We must first recognize that there is a distinction between the "network management" of monitoring and capacity planning from the "network management" of troubleshooting. For convenience, we'll refer to the former simply as "network management" and the latter as "troubleshooting."

Nearly 100% of network management occurs when networks are not failing.

When today's networks break, it is usually due to either a hard connectivity failure or a routing failure. In either case, neither TCP nor UDP get through.

Error bursts and congestion failures do occur, but these tend to be transient, and whether performed by the TCP engine or in a network management station's SNMP retry logic, the packets do tend to get through eventually. It is interesting that with TCP's congestion avoidance algorithms, TCP based streams behave in a way more likely to alleviate the congestion than unregulated UDP streams.

Quality of service controls (such as RSVP [1]) are coming to the Internet. We expect management traffic will get the ability to request priority. This will help ensure that as long as a pathway exists, there will always be a way to monitor and control the net no matter how congested it gets.

Troubleshooting is a distinct branch of network management and requires tools and techniques quite different from those used for continuous monitoring and control. In troubleshooting, SNMP is, at best, a tertiary level tool with value rather below that of *ping, traceroute nslookup,* and *mtrace.*

The Myth of The Dumb Agent

How often have we been told that agents are simple-minded devices that can't support anything other than a simple SNMP agent? Even if that were true nine years ago, an assertion to which our experience speaks to the contrary, it is completely untrue today.

Today's network devices often contain processors and memory exceeding that of our management platforms of a few years ago. Already these devices perform numerous autonomous operations and have considerable protocol stacks already in place.

Today's network devices are capable of managing themselves, if given the opportunity. (We must admit, however and unhappily, that there is a very large class of price sensitive devices in which every corner that could be cut was cut, including the time to read the relevant specifications or perform any interoperability testing.)

Next stop where?

So where should we take network management? The next sections discuss a few ideas, ranging from the incremental to the radical.

Meta variables

The "Meta-Variable" concept has been around for at least the last six years. It is simple to do and requires no changes to existing protocols or agent implementations.

A meta-variable is simply a MIB variable which exists only in the MIB definition document. Each meta-variable is defined as a function of real MIB variables.

Meta-variables would be used by MIB designers to express useful derivations that can be made from the raw data. This could capture a significant body of empirical knowledge which today is rarely, if ever, recorded.

The function may be simple, such as the dividend produced when an error counting `Gauge` variable is divided by `sysUpTime`. In this case, the result would be an average error rate.

continued on next page

Towards Useful Management *(continued)*

Or the function may be more complex, like something that takes the second derivative with respect to time of that error counting `Gauge`. This function would highlight significant changes in the error rates on an interface, which is a far more useful indicator of trouble than an average error rate.

To reify these meta-variables, a management station would have to perform the function. This implies that the function must be expressed by some procedural statement that can be mapped down to basic SNMP `get` and `set` primitives and polling. One might say that the functions would be best expressed as simple scripts.

The definition of these meta-variables and the functions used to generate them would be expressed in standard MIB definition documents with appropriate formalities so that they could be machine parsed and utilized by a management station.

An extension of the meta-variable concept is to place intermediary devices in the network whose role is to compute these meta-variables and export them as real SNMP variables in a MIB specific to those intermediary devices. Another extension is for the SNMP agents themselves to compute the meta-variables, in which case they become real-variables.

Embedded management applications

The World-wide Web is everywhere. Everybody has a browser. These browsers are a standard user interface available to any application which chooses to communicate using the Web's native protocol, HTTP, as specified in RFC 1945 [2].

Although SNMP itself is relatively "simple," it takes some work to build the MIB support in an agent, and considerably more work to build the management support to utilize the MIB data, and a great deal of work to deploy the manager onto the various network management "platforms."

An HTTP/HTML management server embedded in a managed device, with underlying TCP is not significantly more complex or memory intensive than an SNMP agent with mechanisms supporting generalized lexi-ordering and arbitrary collections of objects in a `set`. (It is easy to vastly underestimate the amount of work required for an agent to handle an arbitrary collection of proposed values which may arrive in a `set` request.)

If one looks at many of today's workstation-based management platforms, one quickly realizes that they are really not much more than a collection of device-specific add-ons.

Those add-ons could be just as easily created by having a device export highly device specific Web pages with controls and user interface paradigms. For example, management platforms take pride in the fact that they can project a rendering of a managed device, so that the operator can point at a port to invoke a control panel for that specific port. This is pretty routine stuff for a typical Web server.

The device vendor ships one, self contained product. That product includes its own management functions and does not depend on anything except that WWW browsers are reasonably uniform and ubiquitous. With respect to its management functions, the vendor controls the horizontal and it controls the vertical; the vendor controls everything about the device and its management, from operation to GUI. It's an extremely attractive proposition.

The great drawback of this approach is that it requires human intelligence to comprehend the WWW forms presented by a device. If one accepts the proposition, as we do, that in the long-term, networks should perform significant self-management, then this approach represents a substantial danger that we will end up further from our goal rather than closer.

Using HTTP as an access method

SNMP should not be confused as being network management. Rather SNMP is merely an access method used by a management station to read and write items in an agent's MIB.

The myths of "The Collapsing Network" and "The Dumb Agent" have forestalled many efforts to consider a connection-oriented alternative to SNMP.

Today's Internet is successfully carrying an enormous transaction load using the World-wide Web's HTTP, which is a TCP-based protocol. HTTP transactions follow a very simple life-cycle:

- Client creates a TCP connection to the server.
- Client transmits an HTTP operation, usually a GET or a POST, to the server. Although both can be used to carry additional information from the client to the server, POST has no restrictions on the size or structure of that information.
- The server responds with an HTTP header followed by a MIME-typed chunk of binary data of arbitrary size. This data may be literally anything that can be reduced to binary. It may be the familiar HTML of Web pages, a JPEG image, or instructions to the browser on how to launch an MBONE viewer.
- The connection is closed.

HTTP's major shortcoming is that it doesn't do enough work per TCP connection. Efforts are underway to reduce this weakness.

One could readily conceive of a number of ways to encode MIB information in that chunk of binary data. It could be truly binary, with its own MIME type. Or it could be embedded in HTML as readily identifiable, machine parseable, structured comments.

One might think that the real issue with this approach is how to map `get`, `get-next`, `get-bulk`, and `set` onto this scheme.

However, the real issue is whether we really need the `get*` trinity at all. The `get` operation is the only silver-bullet of the three SNMP retrieval operations; the latter two are merely means to get past SNMP's limited data unit imposed by the myth of "The Collapsing Network." As such, all three retrieval operations could be collapsed into a single `get-subtree` operator that takes a single parameter, an object identifier, and returns all objects which are prefixed by that OID. For convenience, we ought to define the subtree traversal to return the objects in lexicographic order; and, for efficiency, we should allow a list of prefixes and allow the return of multiple sub-trees.

So, how would this actually be mechanized over HTTP?

Consider SNMP queries as the equivalent of a WWW form in which the user or management station simply lists the MIB objects it wants to obtain or set values into. The SNMP response would be the Web page returned as a result of processing the input form.

continued on next page

Towards Useful Management *(continued)*

For processing efficiency, this result need not be encoded in a way that could be directly presented to a human user. The data could be handled either by a special application which speaks HTTP or by a management plug-in to a WWW browser.

One very attractive feature about this approach is that it may be able to piggyback on those WWW security features which are falling into place.

The main drawback of this scheme is that it can be highly intensive in its use of TCP connections, but as has been mentioned, the WWW community is already facing and, hopefully, resolving this problem.

It has been argued by some that this approach would degenerate into a prodigious number of short TCP connections, each retrieving only a small number of MIB variables. This is a valid concern. It has also been argued that the gain offered by TCP is not so great when comparing with the `get-bulk` operator. This is true, however, `get-bulk` is not widely deployed (yet). And it merely changes the point at which the curve of TCP efficiency crosses that of SNMP efficiency; it does not change the fact that, as MIB retrieval size increases, TCP becomes more efficient than UDP-based SNMP.

Using long-lived SNMP Associations

Consider the proposition that there exists a long-term relationship between a management station and managed devices on the network.

In SNMP, this relationship is somewhat vague and tends to be indirectly visible as polling by managers (to determine ongoing device status), `trap` destination configuration in agents, and table management in RMON devices. In the various SNMPv2 proposals, this relationship was made manifest through the various administrative frameworks.

Why not go the next step to acknowledging the relationship and creating an explicit manager–agent "association"? This association would be composed of security and other state information and there would exist, whenever possible, an open transport connection between the manager and agent. (When that underlying transport connection fails, the two ends would attempt to reconstruct it and re-synchronize their association state.)

This approach vastly simplifies the issues of security—authentication and privacy exchanges would occur at association startup and would be cross-checked at important points in the association (typically in the form of a handshake when re-building transport connections and as cryptographic-checksums embedded as integrity checks in the various transactions crossing the association.)

This approach also obtains a significant performance improvement over today's SNMP when moving any significant amount of data. (With today's TCP protocol engines for small queries, however, there may be three or four packets crossing the net rather than the two for UDP based SNMP, although the comparative analysis can be rather complex and highly subject to packet loss rates and the TCP windowing and ACK behavior of a given TCP implementation.)

Scripting MIBs

Through the use of a MIB one can insert a script into a device, start its execution, poll for completion (or await a `trap`), and fetch the results.

One can imagine, for example, a script that monitors the variables in a device watching for tell-tale signs, such as a rapid increase in an error rate. The script could then either report the problem, trigger additional diagnostic tests, or take corrective action. (The latter two would require sophisticated scripts.) Scripts are ideal mechanisms to evaluate and act upon meta-variables, as described earlier.

Scripts are often expressed in a simple interpretive language. Each line of the script is simply a row in a table of octet string variables.

This is not a new idea: some years ago David Levy of SNMP Research published a "Script MIB" and the University of Delft allows a management station to inject Scheme language programs as RMON filters.

The real difficulties of all script approaches are not the scripts themselves or the language used (although, as one can expect, there are competing camps advocating *tcl, Scheme, Java, Python, APL, RPG, COBOL,* or *French.*)

The difficulties are these:

- *Script security:* Can the script be kept within bounds? This is a difficult issue because, almost by definition, network management implies the exercise of discretionary control. If network management is to have the ability to make beneficial changes, it almost necessarily has the power to cause damage if misused.
- *Script integrity:* One wants to be sure that the script being executed is actually the one intended. In Java, there are already authorities who will place their imprimaturs on a script with the guarantee that "this script is safe" according to some criteria.
- *Resource control:* Scripts are programs and as such they can consume memory and computing cycles and potentially other resources. How does one put a quota on a script?
- *Script control:* A management station needs to be able to take control of an executing (or run-away) script. There needs to be a way to halt scripts.
- *Script recovery:* A script can have a lifetime longer than the memory of the management station which started it. It is important that a management station can learn of the existence of scripts it created.
- *Expressive power:* There is a great deal of room for differences of opinion regarding the fundamental actions which a script can invoke. Our own experience is that the primitives should be reasonably high level and should include the following:
  - ICMP *ping* (with control over packet size, packet contents, IP options, and retry intervals). This *ping* should capture round trip times, loss rates, data inconsistencies between what is sent and what is returned, and any ICMP unreachable messages. On multi-homed machines, the script should have control over which interface is used to send the packet.
  - *Traceroute* (with control over source routing, packet sizes, retry intervals, and maximum and minimum TTL).
  - Path MTU discovery.
  - DNS lookup tools.
  - SNMP operations.

continued on next page

Towards Useful Management *(continued)*

- *Script migration:* With a script MIB, the migration of scripts from one machine to another is not an issue, since the management station creating the script controls the migration.
- *Script debugging:* Scripts are programs and programs have bugs. Initially we can expect scripts to be fairly simple and amenable to simple debugging techniques. However, as scripts grow in complexity, we will need means to trace their execution, trap exceptional conditions, set breakpoints, and inspect script variables.

Initially we can expect scripts to be fairly simple: a good first step might simply be to watch what human managers do and use the scripts as simply macros for commonly executed sequences. Over time, as experience grows, scripts should grow in sophistication and, as we learn to trust them, given more power to take limited actions without asking for human permission first. This leads us to the next step:

Semi-autonomous Area Managers

The notion of scripts opens up the possibility that one can design a system to delegate the monitoring and control tasks from a high level manager to subordinate "area managers" in close proximity to those devices that they are managing.

This is not a new idea. Professor Yechiam Yemini of SMARTS [3] has been building tools using these techniques for many years. Java's popularity is extending this idea to areas other than network management.

The basic idea is "management by delegation," the superior level manager creates a script which it downloads into the area manager for execution. The area manager is, of course, a multi-threaded device and can execute many scripts simultaneously, perhaps on behalf of multiple superior managers.

An area manager would usually be given authority over devices with which it has inexpensive, low latency communications. One might conceive of an area manager's span of control as a single LAN or a group of LANs connected with a single high performance router.

The area manager might interact with the end devices using scripts, but it is far more likely that it would be done using traditional SNMP. One of the benefits of the proximity of the area manager and the ultimate devices is that the high bandwidth and presumably low packet loss rate would allow SNMP exchanges to be done rapidly and with minimal data distortion due to non-atomic snapshots of device tables.

An interesting possibility of area managers is that if they are equipped with out-of-band communications paths they can play a very useful role in network troubleshooting. Anyone who has ever repaired networks knows that you always need to be in at least two places at once. An area manager running a pre-loaded script can act as a troubleshooter's remote eyes and ears. For example, an area manager might be running a script which says:

> Watch the network traffic and routing protocols and periodically *ping* sites off the local net to confirm outside connectivity. Should outside connectivity fail, perform *traceroutes* and report the results using the out-of-band channel.

Network management Worms

The term "worm" in networking comes from the John Brunner's book *The Shockwave Rider,* and refers to a program that moves about a network, from computer to computer. It has become a rather pejorative word due to the widely-reported Internet virus of November 3, 1988. However, worms are potentially very valuable. For example, many years ago at Xerox PARC there were worms that propagated through the facility's computers at night to perform diagnostics on otherwise non-busy workstations.

In terms of network management, worms are really scripts that can replicate and migrate. They are really just the next step in the continuum that begins with script MIBs. Some researchers in the network management community are already working with migratory programs. They can perform network device discovery using a worm that migrates through the network and sends a report back to a central logging address whenever the worm moves to a new machine.

Effective use of worms requires that they be "safe," that they have finite lifetimes and limited appetites for network and computing resources, and that they can themselves be located, managed, and terminated.

Summary and conclusion

In this article we have illustrated a few ways that network management can become something better than it is today. We have taken a rather opinionated position, not because we believe we are right (although we hope we are), but rather to try to ignite new work in network management. None of the ideas presented here are impossible. Any one could be developed and deployed within 12 months.

References

[1] `http://www.isi.edu/div7/rsvp/rsvp.html`

[2] T. Berners-Lee, R. Fielding, H. Nielsen, "Hypertext Transfer Protocol—HTTP/1.0, RFC 1945, May 1996.

[3] `http://www.smarts.com/company.html`

[4] `http://www.mathcs.carleton.edu/students/darbyt/pages/worm.html`

[5] `http://www.mathcs.carleton.edu/students/darbyt/pages/history.html`

[6] Stallings, W., "Back to Basics: The Hypertext Transfer Protocol (HTTP)," *ConneXions,* Volume 10, No. 8, August 1996.

[7] Stallings, W., "RMON 2: The Next Generation of Remote Network Monitoring," *ConneXions,* Volume 10, No. 5, May 1996.

[8] `http://www.simple-times.org/pub/simple-times/issues`

Ed.: This article first appeared in the July 1996 issue of *The Simple Times* [8]. Used with permission.

CHRIS WELLENS holds a B.A. from Wellesley College, and a M.S. from the University of Southern California. She is the CEO of InterWorking Labs, a developer of SNMP test suite software and agent simulators that she co-founded in 1993. Chris was Director of Technology for Interop Company, and held senior positions in marketing and engineering with Sun Microsystems. Chris is a frequent speaker at networking conferences on topics ranging from Internet commerce to Web marketing and distribution to network management. Chris is also a contributing writer at *LAN Magazine,* and Instructor in Software Product Marketing at UC Berkeley. Chris is a member of the IETF. She can be reached at `chrisw@iwl.com`

KARL AUERBACH holds an A.B. from the University of California Berkeley and a J.D. cum laude from Loyola University Law School. Karl is Principal Software Engineer at Precept Software, a developer of standards-based (RTP/RTPC and RSVP), real-time, multimedia networking solutions. Prior to Precept, Karl architected and implemented Dr. Watson, the Network Detective's Assistant, a product providing active diagnostic capabilities for TCP/IP networks. Karl founded Epilogue Technology where he designed the first commercial SNMP engine. Karl is a member of the IETF, the State of California Bar Association, and is a past member of the InteropNet NOC team. He can be reached at `karl@precept.com`

The Clipper Proposal

by
Charlie Kaufman, Iris Associates,
Radia Perlman, Novell, Inc.,
and Mike Speciner, Color-Age Inc.

Introduction

The U.S. government has proposed technology to preserve its ability to wiretap otherwise secure communication. To do this it must either prevent use of encryption, break the codes used for encryption (as it did in a military context during World War II), or somehow learn everyone's cryptographic keys. The proposed technology takes the last option. It allows the Government to reconstruct your key (only upon court order and with legitimate cause of course). This is made possible through the use of a device known as the Clipper chip. A lot about Clipper is classified by the government as secret (and classified by a lot of other people as evil). The simple concept is that encryption is done with a special chip (the Clipper chip). Each chip manufactured has a unique key, and the government keeps a record of the serial number/encryption key correspondence of every chip manufactured. Because not all people have complete trust in the government, rather than keeping the key in one place, each key is broken into two quantities which must be XOR'ed in order to obtain the actual key. Each piece is completely useless without the other. Since each piece is kept with a separate government agency, it would require two U.S. government agencies to cooperate in order to cheat and obtain the key for your Clipper chip without a valid court order. The government assures us, and evidence of past experience supports its claim, that cooperation between U.S. government agencies is unlikely.

The Clipper proposal is controversial, starting with its name. The name will certainly change, since it violates someone's trademark on something unrelated. But since everyone calls the proposal Clipper, we will too, especially since the new name has not been chosen.

Clipper benefits

Why would anyone use Clipper when alternative methods should be cheaper and more secure? The reason alternatives would be cheaper is that enforcing the ability of the U.S. government to wiretap will add a lot of complexity. Proponents of Clipper have given several answers to this question:

- The government will buy a lot of Clipper chips, bringing the cost down because of volume production, so Clipper will wind up being the most cost-effective solution.
- Encryption technology is only useful if both parties have compatible equipment. If you want to talk securely to the U.S. government, you will have to use Clipper. So any other mechanism would have to be implemented *in addition* to Clipper.
- Again, since encryption technology is only useful if both parties have compatible equipment, if Clipper takes over enough market share, it will essentially own the market (just like VHS, a technically inferior standard supposedly, beat out Beta in the VCR marketplace). Since Clipper will be one of the earliest standards, it might take over the marketplace before any other standards have an opportunity to become entrenched. Most people won't care that Clipper enables wiretapping, because they'll assume they have nothing to fear from the U.S. government wiretapping them.
- The government claims that the cryptographic algorithm in Clipper is stronger than you'll be able to get from a commercial source.

Civil libertarians fear Clipper is a first step towards outlawing untappable cryptography. Clipper proponents say it is not. It's true that outlawing alternatives is not part of the Clipper proposal. However, there have been independent efforts to outlaw cryptography. Those efforts have been thwarted in part with the argument that industry needs security. But if Clipper is deployed, that argument goes away.

Clipper is designed for telephones, fax, and other low-speed applications, and in some sense is not relevant to computer networking. Many people regard it, however, as a first step and a model for taking the same approach for computer networks. Telephones are easy to tap, and cellular telephones make eavesdropping even easier. Because of this, people have developed encrypting telephones, which so far have been too expensive to catch on, but today it is technically feasible to make encrypting telephones inexpensively. This makes the Government nervous because criminals are sometimes convicted using evidence gathered through wiretaps. To fill the need for encryption without the U.S. government giving up the ability to wiretap (with legitimate reason and through a court order), the U.S. government has proposed the Clipper chip. The Clipper proposal offers high-grade encryption while preserving the ability of the U.S. government to wiretap.

How it works

Technically the concept of Clipper is reasonably simple. Each Clipper chip manufactured contains a unique 80-bit key and a unique 32-bit ID. For each key K, an 80-bit random number K_1 is selected, and then $K_2 = K \oplus K_1$ is computed. ($\oplus$ is the XOR function). Neither quantity K_1 nor K_2 gives any information about K, but if you know both K_1 and K_2, you can compute K easily, since it is $K_1 \oplus K_2$. K_1 (and the unique ID) is given to one federal agency, and K_2 (and the unique ID) is given to a different federal agency. It hasn't been decided which agencies they'll be, but they should be agencies that everyone trusts, like the IRS, and that are independent of one another without likelihood of collusion, like the Army and the Navy.

Let's say that Alice, using a telephone containing a Clipper chip, wants to talk to Bob, who has a similar device. Alice's chip has been registered with the two agencies (as has Bob's). Let's say Alice's chip has unique ID ID_A and secret key K_A. With a court order, the government can obtain the two components of Alice's chip's key and then reconstruct K_A. Without a court order, K_A remains secret.

What key will Alice and Bob use for communicating? It can't be K_A or K_B (Bob's chip's secret key) because neither side wants to reveal its secret key. So Alice and Bob use some mechanism, unspecified in the Clipper standard, to produce a shared secret key S. A reasonable choice is Diffie-Hellman [1] which in fact has been implemented in the current crop of Clipper phones. Alice feeds S to her Clipper chip and Bob feeds S to his Clipper chip. The chips use S to encrypt and decrypt the data. So where does K_A come in, and how would the government, knowing K_A, be able to decrypt the conversation? Also, how does the government know the unique ID of Alice's chip in order to obtain (with court order) K_A? It would be an administrative nightmare for the U.S. government to try to keep track of who owns each Clipper chip.

LEAF

The information the government needs is in a field known as the LEAF, for *Law Enforcement Access Field,* that Alice's and Bob's Clipper chips transmit along with the encrypted data. Although Bob's chip does not utilize any of the information in the LEAF it receives from Alice's chip, Bob's chip refuses to communicate unless it receives a valid-looking LEAF.

continued on next page

The Clipper Proposal *(continued)*

The central challenge in the design of Clipper is preventing someone from building a device that uses a Clipper chip for secure communication but substitutes garbage for the LEAF before transmitting the data.

The Clipper design is very clever. The LEAF that Alice transmits to Bob contains ID_A, $K_A\{S\}$ (in our notation, $K_A\{S\}$ means K_A encrypted with the key S), and a checksum C. The field ID_A enables the government to retrieve K_A and then decrypt the field $K_A\{S\}$ to obtain S. Since the same key S is used in both directions, the government only needs one of the keys K_A or K_B in order to decrypt the entire conversation.

How does Bob's chip know whether the LEAF is valid? Bob's chip can't know whether ID_A is the correct value and it can't decrypt $K_A\{S\}$ to check if it's the correct encrypted key. Bob's chip makes its decision based on the value of the field C. C is basically some sort of message digest of the other fields in the LEAF (ID_A and $K_A\{S\}$) and the key S. Bob's chip computes the message digest of the values in the received LEAF for ID_A and $K_A\{S\}$, along with S (which Bob's chip knows). If the computed checksum matches the received C, then the entire LEAF is assumed to be valid.

The message digest algorithm used to compute C need not be secret, so what's to prevent someone from foiling the government's ability to wiretap by modifying the quantity $K_A\{S\}$ (or ID_A) and sending the matching C? To prevent modification of the LEAF, all Clipper chips share an 80-bit secret *family key, F*. The quantity $ID_A \mid K_A\{S\} \mid C$ (in our notation, that means the three quantities ID_A, $K_A\{S\}$, and C concatenated) is encrypted with F. Since Bob's chip knows F, it can decrypt the LEAF to extract ID_A, $K_A\{S\}$, and C. It can't verify that either ID_A or $K_A\{S\}$ is correct, but knowing S, it can compute C. If the C in the LEAF doesn't match the computed C, the chip will refuse to decrypt the conversation.

With the LEAF thus guaranteed, the government can (with a valid court order) tap Alice's line, read ID_A from the LEAF, and then retrieve K_A. Then it can read $K_A\{S\}$ from the LEAF and decrypt it to obtain S. At this point, it can decrypt the recorded conversation.

SKIPJACK

It is rather astonishing that all Clipper chips will know the value F, and yet the expectation is that the value will remain secret. Clipper chips are carefully manufactured so that it should not be possible, by taking one apart, to obtain F. The encryption algorithm (known as *SKIPJACK*) is also supposed to be kept secret, and the same technology that prevents someone from reverse-engineering the encryption algorithm will protect F.

How much of a disaster would it be if someone discovered F? It would not make Clipper-protected conversations less secure. However, it would mean someone could build a device that interoperated with Clipper devices and foiled wiretapping by generating garbage for $ID_A \mid K_A\{S\}$, computing the proper value for C based on the garbage and the session key S, and encrypting it all with F.

It has been pointed out that people could, with less effort, build a device that encrypted the data before transmitting it to the Clipper chip. Provided the receiver had a compatible device (Clipper plus the extra encryption device), the output stream would look like normal Clipper output, until someone, under court order, decrypted the conversation and realized they were obtaining ciphertext.

Why use Clipper?

If you had a device capable of encrypting conversations, why would you bother going through the extra step of sending the encrypted stream through a Clipper chip? Perhaps you don't have complete faith in your cryptographic algorithm. Another reason surfaces if it becomes illegal or suspicious to use non-Clipper encryption. If you send your encrypted stream through a Clipper chip, the government would not know that you were using your own encryption, until after it obtained the court order and decrypted your conversation. One wonders what penalties could be associated with using non-Clipper encryption that would deter the sorts of people who are worried about being wiretapped.

> Prosecutor: "Did you murder your wife?"
> Defendant: "No."
> Prosecutor: "Do you know the penalty for perjury?"
> Defendant: "No, but I bet it's less than the penalty for murder!"
>
> —*Classic joke*

If someone discovered F and built a Clipper-compatible device, then Alice, with that device, could foil wiretapping while talking to Bob, who had an ordinary Clipper device. If the government had Alice under suspicion, and therefore had obtained her Clipper key, it would still need to get a court order for Bob's Clipper key before it could decrypt the Alice–Bob conversation. If both Alice and Bob had one of these Clipper-compatible devices, then the government would discover, only after court order, that Alice and Bob were not using real Clipper devices, just as it would discover if Alice and Bob were using an extra encryption step before the Clipper chip.

Real secrets

Matt Blaze [2] discovered an interesting property of the Clipper design that makes it possible for Alice, with a lot of effort, to forge a LEAF in a way that Bob will accept, but will prevent government wiretaps. The problem is that the quantity C is only 16 bits long. Remember that Bob's chip cannot verify that either of the quantities ID_A or $K_A\{S\}$ is correct. All the chip can do is verify that C is based on those quantities and the proper key S.

The LEAF is 128 bits long, consisting of 32 bits of ID, 80 bits of encrypted key, and 16 bits of checksum. Bob's chip will accept any sequence of bits for the ID and the encrypted key just so long as those fields, plus the Alice–Bob session key, produce the correct 16-bit checksum. The implication of this is that if Alice were to send a 128-bit random number, it would have 1 chance in 2^{16} of having the correct checksum, when decrypted with the family key F. Now we get to Matt Blaze's attack. Alice can use any Clipper chip as a tester. She tells it she wants to converse using the key she's agreed upon with Bob. Then she feeds it random 128-bit LEAF values. On average she'll only have to try 2^{15} random numbers before one will wind up with the correct checksum.

Now that she knows a LEAF value that will be accepted by Bob's chip for the key she's sharing with Bob, she needs a special piece of hardware that takes the output stream from her Clipper device, removes the LEAF values it transmits, and substitutes the one she found. Bob's device won't know there's a problem, and the problem will only be detected when the government attempts to wiretap.

Matt Blaze's attack is not important in practice, since it would either take a lot of parallel hardware or cause an unacceptable delay in conversation startup, but the discovery was useful in embarrassing the designers for having missed such an "obvious" flaw.

continued on next page

The Clipper Proposal *(continued)*

[Ed.: This article is exerpted from the book *Network Security: Private Communication in a Public World,* by Charlie Kaufman, Radia Perlman, and Mike Speciner, ISBN 0-13-061466-1, Prentice Hall, 1995. Used with permission.]

References

[1] Diffie, W., and Hellman, M. E., "New directions in cryptography," *IEEE Transactions on Information Theory,* Volume 22, No. 6, 1976, pp. 644–654.)

[2] Blaze, M., "Protocol Failure in the Escrowed Encryption Standard," Proceedings of the Second ACM Conference on Computer and Communications Security, November 1994.

[3] Chapman, D. B., and Zwicky, E. D., "Internet Security Strategies," *ConneXions,* Volume 9, No. 12, December 1995.

[4] Chapman, D. B., and Zwicky, E. D., "Internet Security Policies," *ConneXions,* Volume 10, No. 1, January 1996.

[5] Stallings, W., "Cryptographic Algorithms," Part I: Conventional Cryptography, *ConneXions,* Volume 8, No. 9, September 1994. Part II: Public-Key Encryption and Secure Hash Functions, *ConneXions,* Volume 8, No. 10, October 1994.

[6] Stallings, W., "Pretty Good Privacy," *ConneXions,* Volume 8, No. 12, December 1994.

[7] Doty, T., "The Firewall Heresies," *ConneXions,* Volume 9, No. 6, June 1995.

[8] Kaliski, B., "An Overview of Public-Key Cryptography Standards," *ConneXions,* Volume 6, No. 5, May 1992.

[9] B. Clifford Neuman and Theodore Ts'o, "Kerberos: An Authentication Service for Computer Networks," *IEEE Communications,* Volume 32, No. 9, September 1994.

[10] Levy, Steven, "Clipper Chick," Interview with Dorothy Denning, *WIRED,* 4.09, September 1996, `www.wired.com/4/09/denning`

[11] "IAB and IESG Statement on Cryptograhic Technology and the Internet," *ConneXions,* Volume 10, No. 8, August 1996.

RADIA PERLMAN currently works at Novell, Inc., where she designs distributed algorithms and protocols for use in routing, database synchronization, and security. She is on the IAB, the advisory board for the IETF, which standardizes protocols for the Internet. Before Novell she was the routing architect at Digital Equipment Corp., where she designed the spanning tree algorithm used by bridges, and DECnet routing, which was standardized by ISO as the IS–IS protocol, and critical portions of which are included in all modern link state routing protocols. She has numerous patents in the fields of routing and security. She has S.B. and S.M degrees in mathematics and a PhD in computer science from MIT. Her thesis was a design for a sabotage-proof network that is practical to deploy.
E-mail: `radia_perlman@novell.com`

MIKE SPECINER is chief architect at Color-Age Inc., a manufacturer of networked *PostScript* color print servers. Besides graphics, he has extensive experience in databases, operating systems, communications, and computer aided software engineering. He holds several patents in the area of computer graphics. He holds S.B. degrees in math and physics from MIT, and did graduate work in computer science, math, and physics at MIT. E-mail: `ms@color-age.com`

CHARLIE KAUFMAN works for Iris Associates as Security Architect for Lotus Notes. He participates in a number of IETF standards efforts and is chair of the Web Transaction Security working group. Previously, he was Network Security Architect for Digital Equipment Corporation. He holds over 20 patents in the fields of computer security and computer networking. He holds a B.S. from Bates College and an M.A. from Dartmouth College, both in Mathematics. He can be reached as: `charlie_kaufman@iris.com`

Opinion: Participatory Speech Wins

by John S. Quarterman, Texas Internet Consulting

As Judge Dalzell put it:

> "...the Internet may fairly be regarded as a never-ending worldwide conversation. The Government may not, through the CDA, interrupt that conversation. As the most participatory form of mass speech yet developed, the Internet deserves the highest protection from governmental intrusion."

Introduction

On 11 June 1996, three judges of the U.S. District Court for the Eastern District of Pennsylvania decided for the plaintiffs against the federal government over the so-called *Communications Decency Act* (CDA). This decision goes well beyond the best many of us had expected from a court decision, because these judges did not attempt to equate the Internet with some previous form of communication; they explicitly recognized and enumerated the differences, for example in Judge Dalzell's words:

> "Four related characteristics of Internet communication have a transcendent importance to our shared holding that the CDA is unconstitutional on its face. We explain these characteristics in our Findings of fact above, and I only rehearse them briefly here. First, the Internet presents very low barriers to entry. Second, these barriers to entry are identical for both speakers and listeners. Third, as a result of these low barriers, astoundingly diverse content is available on the Internet. Fourth, the Internet provides significant access to all who wish to speak in the medium, and even creates a relative parity among speakers."

This decision is a great victory for the Internet, for the Constitution, and for the free flow of information worldwide. It is good even for those who opposed it. We should celebrate it. Then we should act to protect it.

Text of the Decision

The text of *ACLU v. Reno* is long, but well worth reading, and has been made available in convenient HTML format [1], by Bob Bickford [2], who has even prepended a table of contents.

My part of the "et al." was as one of the *Association of Publishers, Editors, and Writers* (APEW) [3]. If you never heard of APEW before, that's no surprise; it was made up for this purpose when our original suit, led by the American Library Association (ALA), was merged with the ACLU one.

The Judges view the Internet

One distinction between judges and legislatures is that judges are trained to know legislative history, and to review laws in the light of it. For example, Chief Justice Sloviter writes:

> "The government makes yet another argument that troubles me. It suggests that the concerns expressed by the plaintiffs and the questions posed by the court reflect an exaggerated supposition of how it would apply the law, and that we should, in effect, trust the Department of Justice to limit the CDA's application in a reasonable fashion that would avoid prosecution for placing on the Internet works of serious literary or artistic merit. That would require a broad trust indeed from a generation of judges not far removed from the attacks on James Joyce's *Ulysses* as obscene."

continued on next page

Participatory Speech Wins *(continued)*

This court was faced with two descriptions of the Internet that did not appear to be remotely the same thing. Was the Internet a sewer of degenerate filth thrust upon an unwitting and unwelcoming public, particularly upon innocent children? Or was it the greatest boon to interpersonal communications, research, and learning ever known to humanity?

The judges chose to investigate it directly themselves (which is something that the great majority of Congress members never did). They learned what the Internet is, and they got it right. They avoided common mistakes, such as equating the Internet with a single wide area network, or with a single service, such as WWW or electronic mail, although they occasionally stumbled over topics such as the nature of the *Domain Name System* (DNS). They understood the Internet's size, growth rate, international extent, and varied means of interpersonal communication:

2 "Some networks are "closed" networks, not linked to other computers or networks. Many networks, however, are connected to other networks, which are in turn connected to other networks in a manner which permits each computer in any network to communicate with computers on any other network in the system. This global web of linked networks and computers is referred to as the Internet."

4 "These communications can occur almost instantaneously, and can be directed either to specific individuals, to a broader group of people interested in a particular subject, or to the world as a whole."

74 "...It is no exaggeration to conclude that the content on the Internet is as diverse as human thought."

80 "It follows that unlike traditional media, the barriers to entry as a speaker on the Internet do not differ significantly from the barriers to entry as a listener. Once one has entered cyberspace, one may engage in the dialogue that occurs there. In the argot of the medium, the receiver can and does become the content provider, and vice-versa."

81 "The Internet is therefore a unique and wholly new medium of worldwide human communication."

Pornography on the Internet?

The judges searched for pornography and found it was there, but in nowhere near the huge proportions of networked content erroneously reported by *Time Magazine* [4] and others.

83 "There is no evidence that sexually-oriented material is the primary type of content on this new medium. Purveyors of such material take advantage of the same ease of access available to all users of the Internet, including establishment of a web site."

86 "Once a provider posts its content on the Internet, it cannot prevent that content from entering any community. Unlike the newspaper, broadcast station, or cable system, Internet technology necessarily gives a speaker a potential worldwide audience. Because the Internet is a network of networks (as described above in Findings 1 through 4), any network connected to the Internet has the capacity to send and receive information to any other network..."

88 "Communications over the Internet do not "invade" an individual's home or appear on one's computer screen unbidden. Users seldom encounter content 'by accident.'..."

The judges noted that there is already a rating system for the web, PICS, or *Platform for Internet Content Selection* [5], coordinated by the World-Wide Web Consortium. They noted that there are already end-user filtering packages for the Internet, eight of which they cited by name:

- Cyber Patrol [6]
- CYBERsitter [7]
- The Internet Filter
- Net Nanny [8]
- Parental Guidance
- SurfWatch [9]
- Netscape Proxy Server [10]
- WebTrack [11]

The CDA shredded

Finally, the court tore CDA into little shreds. It dismissed concerns of child pornography or obscenity because there are already laws against those things. As Judge Dalzell put it:

> "Indeed, the Government could punish these forms of speech on the Internet even without the CDA."

The court denied the applicability of the Pacifica "Seven Dirty Words" precedent because that one was about broadcast radio, which is uniquely accessible to children, and also subject to regulation because of the limited nature of channels in a broadcast spectrum. As Judge Dalzell wrote:

> "I draw two conclusions from the foregoing analysis. First, from the Supreme Court's many decisions regulating different media differently, I conclude that we cannot simply assume that the Government has the power to regulate protected speech over the Internet, devoting our attention solely to the issue of whether the CDA is a constitutional exercise of that power. Rather, we must also decide the validity of the underlying assumption as well, to wit, whether the Government has the power to regulate protected speech at all. That decision must take into account the underlying technology, and the actual and potential reach, of that medium. Second, I conclude that Pacifica's holding is not persuasive authority here, since plaintiffs and the Government agree that Internet communication is an abundant and growing resource."

The court pointed out that neither "indecent" nor "patently offensive" were defined. Some of the judges strongly criticized CDA for vagueness, particularly in the light of its attempt to impose criminal penalties, which bring in Fifth Amendment Due Process concerns. As Judge Buckwalter wrote:

> "If the Government is going to intrude upon the sacred ground of the First Amendment and tell its citizens that their exercise of protected speech could land them in jail, the law imposing such a penalty must clearly define the prohibited speech not only for the potential offender but also for the potential enforcer."

continued on next page

Participatory Speech Wins *(continued)*

The court demonstrated that CDA was far too broad even to be interpreted as applying to commercial pornographers. The court agreed that the plaintiffs had shown irreparable injury. In Judge Sloviter's words:

> "A wealth of persuasive evidence, referred to in detail in the Findings of Fact, proved that it is either technologically impossible or economically prohibitive for many of the plaintiffs to comply with the CDA without seriously impeding their posting of online material which adults have a constitutional right to access."

The court granted the preliminary injunction requested by the plaintiffs against those parts of the CDA that treat "indecency," declaring them unconstitutional under both the First Amendment and the Due Process Clause of the Fifth Amendment.

To those who support the CDA

We are living in a period of probably the greatest religious and social ferment since the Reformation. Many parents are rightly concerned about raising their children in these times. Even those of us who are not parents have many reasons to be concerned. We would rather live in a world of literates than illiterates, of people who can support themselves than of burdens on the state, of people who are capable of reasonably civil social interaction than of sociopaths. Some parents wish to limit what their children can see and hear. Regarding Internet access, software packages for that already exist, including those cited in the decision. They aren't perfect, both because no filter is perfect, and because these filters tend to err on the too-broad side by pulling up some vegetables along with the weeds. But they at least provide a choice among different filters, and you can lobby the various companies that make them until you get one tuned to your needs. With the CDA, you'd have no such flexibility.

Many people supported the CDA on religious grounds, wanting to protect their children from what they had been told was a slough of pornography and child molesters on the Internet. Many of us oppose it just as religiously, knowing that the real character of the Internet is not that, and knowing how prior censorship has historically been a tool of oppression that destroys far more than it cures, and that often bites the hands even of those who wield it. Many of us are parents, and *because* of that oppose the CDA, preferring to preserve parental and personal choice rather than yield it up to the state, which cannot possibly know our families as well as we do.

If you are from Texas, do you want a Massachusetts judge deciding that you are "indecent" for quoting a traditional song ("stepped in what?")? If you are from Massachusetts, do you want a Texas sheriff arresting you for "indecently" writing his deputy "busted your balls"? If you are from Amsterdam, do you want a California D.A. charging you with "indecency" for discussing the customs of your country, such as unrestricted immigration, legal pornography, or legal drugs? If you vote, do you want your candidate for any office drummed out of the race by revelations of some online "indecency" such as advocating a ban on abortion? If you don't want those things, you don't want the CDA. As Justice Sloviter wrote:

> "But the bottom line is that the First Amendment should not be interpreted to require us to entrust the protection it affords to the judgment of prosecutors. Prosecutors come and go. Even federal judges are limited to life tenure. The First Amendment remains to give protection to future generations as well."

To those who oppose the CDA

Don't be complacent. The first reaction of many people on the net when the CDA was first proposed by Senator Exon was "EFF will do something." Well, EFF was apparently too busy with infighting and alienating its former and potential supporters. Fortunately, *Voter's Telecommunications Watch* (VTW) [12] popped out of the woodwork to get the word out to the net and the public. And the *Center for Democracy and Telecommunications* (CDT) [13] surprised me, among others, by transcending its origins as the inside-the-Beltway business-as-usual operators of EFF when it actually tried to do something about the CDA; showing up in Congress and lobbying against it. Even the *American Civil Liberties Union* (ACLU) [14], which only a few years ago couldn't be bothered to take a stand on anything related to cyberspace (that's why Jerry Berman moved from ACLU first to EFF and then to CDT), took a leading role in the legal challenge against the bill once it was passed.

Special mention has to go to Brock Meeks <brock@well.com>, whose *Cyberwire Dispatch* kept after the story like a Rotweiller on a burglar.

And there were Vice President Al Gore, who spoke against the bill, President Bill Clinton, who said he wouldn't enforce parts of it (although arguably not the most important parts), and Speaker of the House Newt Gingrich, who took a stand on principle when the CDA passed the Senate and said it couldn't happen in the House. Well, Gingrich flipped pretty quick, and, according to a letter from my representative, Lloyd Doggett, insisted on passing the Telecommunications Act of 1996 without separate debate on the CDA. Clinton signed the bill anyway, and Gore hasn't had a lot to say against it since then.

The most effective defense against the CDA came from the net; from VTW, surprisingly from CDT, from Brock Meeks, from assorted writers, and especially from all the individuals and organizations who called and wrote Congress about that bill and who joined in the lawsuit that recently succeeded.

I'm not forgetting the few Congress members who did take real and enduring stands against that travesty of legislation, starting with Sen. Patrick Leahy (D-VT), and continuing with Rep. Ron Klink of (D-PA), Rep. Chris Cox (R-CA), Rep. Ron Wyden (D-OR), and others, especially including the brave few who voted against the CDA in the Senate (the House never had an opportunity to explicitly do so). But people from the net gave them the support and materials they needed to work with. It's only too bad they didn't get more of both; they might have won this thing in Congress. Instead, it went to a court decision.

What next?

The court in Philadelphia handed down only a preliminary injunction, the government has already announced it will appeal, and the Supreme Court is the next stop. Chances are the CDA will get a stake through its undead heart there.

But what if the Court lets the CDA take effect? Even in that worst case, the Internet will not die; it's too large, too decentralized, too international, and too robust for that. But that this (or some other) government does have the power to harass the Internet's providers and users. Providers that promise PG-13 content and that practice censorship will be legally favored, but even they will be held hostage to arbitrary interpretation of the actions of any of their staff or users.

continued on next page

Participatory Speech Wins *(continued)*

As Judge Dalzell put it: "...the Internet would ultimately come to mirror broadcasting and print, with messages tailored to a mainstream society from speakers who could be sure that their message was likely decent in every community in the country...[and] where economic power has become relatively coterminous with influence." The Internet will be damaged, the U.S. Republic will be damaged, and the whole world will be the loser. The country that has led the Internet from the beginning will become the greatest impediment to it.

Will the people of the Internet stand up then and fight an uphill battle in the same Congress that passed this law? I don't know. There will be twice as many of them in the year it will probably take to reach a decision; maybe that will be enough. We'll see.

Suppose the Supreme Court makes a decision as good as the one the District Court made, and the CDA is struck from the lawbooks. CDA proponents won't stop then. There will be further bills, either in Congress or in the states, or both. Judge Buckwalter, even though he broadened his position against the CDA from what he said in the Temporary Restraining Order to what he said in the Preliminary Injunction, made a point of remarking:

> "That is to say that I specifically do not find that any and all statutory regulation of protected speech on the Internet could not survive constitutional scrutiny."

We must be sure that we hold up our side of that scrutiny.

There will be attempts to reinterpret existing laws (remember the Comstock Act is still on the books). And if an administration favoring this sort of thing more strongly than the current one gets into the White House, there will be attempts to slip it in by executive decree.

Don't wait for some millionaire, or organization, or politician, or judge to settle this issue for you. The temptations of fame, money, power, and of believing oneself wiser than the rest are all too real, and have already been the making of charlatans and the unmaking of formerly useful organizations. If you want to preserve the beneficial chaos of the Internet, not to mention the First Amendment to the U.S. Constitution, you will have to do it yourself.

Pick a source of information about these topics, such as VTW, CDT, ALA, or ACLU; the long list of plaintiffs in the recent case would be a good menu for such a choice. Write your Congress members and the President, and don't forget your state and local legislators and executives. Ask political candidates how they stand on freedom of speech on the Internet, and take their answers into account when you vote.

The rest of the story

Finally, don't assume the CDA or something like it is the only threat to the Internet or to the free flow of information. The U.S. government for many years has opposed the development and deployment of cryptography adequate to protect communications from snooping by the U.S. government. The Clinton Administration is still promoting variations on the Clipper Chip idea, which would mandate use of only one cryptography standard and would require key escrow, i.e., you'd have to tell the government what your keys are. The government promises not to use them, except of course whenever it feels like it. (See pg. 10).

What about the rest of the Telecommunications Act of 1996, for that matter, not just the CDA part of it? The whole Act was lobbied for intensely by large telecommunications companies; many of them the same ones who would like to take over the Internet provision market.

Do you want to have to get Internet service from only a choice of a very few multinational providers? If not, you would do well to keep an eye on this one.

What about state initiatives to arbitrarily tax Internet Service Providers (ISPs)? Texas recently invented a Texas Infrastructure Fund (TIF) tax and attempted to make each Texas ISP to pay one percent (1%) of gross revenues to it. Only quick action by the nascent *Texas ISP Association* (TISPA) headed that one off at the pass. If it can happen in the second largest state, it can happen in your state, too.

The Internet is new on the political scene, and every government body is going to want to get control over it, just as every nongovernmental body, nonprofit or commercial, is trying to get a piece of the action. If you want to keep your piece of it, you will need to pay attention and take action when necessary.

The Web of error

Thomas Jefferson once wrote: "Error is the stuff of which the web of life is woven." The Internet was built from the actions of a growing multitude of people and organizations, each of their parts weaving into the whole. No single person or organization ever knew quite where it was going, and no single entity controls it. Let's keep it that way.

References

[1] `http://www.well.com/conf/liberty/cda/index.html`

[2] `http://www.well.com/~rab/`

[3] `http://www.mids.org/apew/`

[4] `http://www.zilker.net/swg/time.html`

[5] `http://www3.org/pub/WWW/PICS`

[6] `http://www.microsys.com/cybers`

[7] `http://www.qdeck.com/qdeck/press/isuite.html`

[8] `http://www.netnanny.com/netnanny/`

[9] `http://www.surfwatch.com`

[10] `http://www.netscape.com`

[11] `http://webtrack.webster.com`

[12] `http://www.vtw.org`

[13] `http://www.cdt.org`

[14] `http://www.aclu.org`

[15] Quarterman, J., "*Opinion:* A Statement Against the CDA," *ConneXions,* Volume 10, No. 6, June 1996.

[16] "IAB and IESG Statement on Cryptograhic Technology and the Internet, *ConneXions,* Volume 10, No. 8, August 1996.

JOHN S. QUARTERMAN wrote the first book about the Internet and related networks, *The Matrix: Computer Networks and Conferencing Systems Worldwide,* Digital Press. He is a co-author of five other books, *The Design and Implementation of the 4.4BSD UNIX Operating System, UNIX, POSIX, and Open Systems: The Open Standards Puzzle, Practical Internetworking with TCP/IP and UNIX, The Internet Connection: System Connectivity and Configuration,* and *The E-Mail Companion: Communicating Effectively via the Internet and Other Global Networks,* all from Addison-Wesley. He is Editor of the color *Matrix Maps Quarterly* and the monthly *Matrix News,* both about issues that cross network, geographic, and political boundaries, and both published by Matrix Information and Directory Services, Inc., (MIDS) of Austin, which also conducts demographic surveys and other research into the composition of the Internet and related networks and their users. He is a partner in Texas Internet Consulting (TIC), which consults in networks and open systems, with particular emphasis on TCP/IP networks, UNIX systems and standards. He is a partner in Zilker Internet Park, which provides Internet access from Austin, Texas. E-mail: `jsq@mids.org`

Book Review

On To Java, by Patrick Henry Winston and Sundar Narasimhan, ISBN 0-201-49826-X, Addison Wesley Longman, 1996, 328 pages, 46 chapters.

Structure

This is a very nice book on the new programming language, Java. It is designed around a standard formal course text structure that introduces various features of the language, chapter by chapter, around a single application that is gradually built, and then refined. It is not a reference book for the language (it features no section summarising the syntax or semantics of the language in one place) nor is it a "sit down and read" book on Java, but it would be a very nice undergraduate programming book.

However, it is not a network related book (there is a section on turning Java programs into Applets, and brief description of how to pass parameters from a Web page access through to a Java Applet, but little else—for instance, nothing on the Java Network Class Library, or the attempts to make Java secure, or to distribute it via Joe or CORBA).

Each chapter is divided into "byte sized chunks" each of which addresses a point/hint/language feature, and in a very readable manner.

Language

There are a number of "motherhood" type chapters on Object Oriented design and programming (e.g., on class-instance programming, use of "is-a" and "has-a" for choosing which to use, chapters on modularisation, inheritance etc., etc., iteration compared with recursion beautifully explained, etc.).

The coverage is good of the base language, and of the approach, and Awt and file i/o Class Libraries, although (apart from the net* class) one would like to see a model of the byte code language included, a comparison with scripting languages (such as Java Script itself, and Tcl/Tk, as well as other Object Oriented languages—other Java books have used **C++** or even Visual Basic as a baseline comparison to help programmers with existing language knowledge).

The book could also have made use of the standard Java class documentation, just to help people find their way around things like the File Input Stream etc.

Typos

There are a small number of curious typographical errors where the initial letter of a class or variable name is missing (presumably a TeX mistake or error on input where a font change happens) but the accuracy of the book is largely good. (Examples of typos: missing *z, h* and *v* on pages 69, 77, 163 and 309–311 from variables *z*ero, *h*our and *v*ector).

Summary

This is the best introductory programming book to Java that I have seen, but for completeness, you would need a network/www book, and a more in-depth description of the Java VM, the full current Java release Class Libraries, and the Security Model.

—*Jon Crowcroft, University College London*
J.Crowcroft@cs.ucl.ac.uk

Call for Papers

The May 1997 issue of *IEEE Network* will feature the topic of *Network and Internet Security*. Private and public organizations are increasingly dependent on distributed computer systems and computer networks such as the global Internet. However, the commercialization and escalating popularity of the Internet have been accompanied by a rising level of network-related security attacks despite firewalls, encryption, anti-virus programs, and other security measures.

Today, security attacks can originate from a greater number of sources in more varied forms. New networking technologies (e.g., fiber optics, ATM) enable data to be moved at rates of gigabits/s, where security breaches and data transmissions can occur on timescales much faster than human intervention. In addition, computer technology is being used for increasingly sophisticated, automated tools that allow any non-expert to perpetrate serious security attacks. As the issue of security gains prominent attention in scientific, political, and commercial arenas, it poses a potential hindrance to the continued growth of the Internet and other computer networks.

Topics

Contributions are invited to the feature issue of *IEEE Network* which will represent the current state of the art covering all aspects of network and internetwork security. Topics may include, but are not limited to:

- Cryptography and privacy
- Viruses, worms, and intruding software
- Internet commerce
- Firewalls and access control
- Authentication and digital signatures
- Key management and exchange
- Certificates and certification authorities
- Secure e-mail
- Secure network management

Submissions

Contributions should be submitted by *November 1, 1996* to the guest editor:

Thomas M. Chen
GTE Laboratories, Inc.
40 Sylvan Road
Waltham, MA 02254
USA
Phone: +1 617-466-2758
Fax: +1 617-890-9320
E-mail: tchen@gte.com

Call for Papers

Background

INET '97 "The Internet: The Global Frontiers," will be held June 24–27, 1997 in Kuala Lumpur, Malaysia. INET, the annual meeting of the Internet Society, is the premier international event for Internet and internetworking professionals and the crossroads at which the world's pioneers of cyberspace meet to exchange experiences and plan their next steps. Each year, network technologists, industry and government representatives, and social experts meet to exchange experiences, share information, and shape the future of the Internet and its related internetworking technologies.

INET '97 will address both the traditional and evolving frontiers of the Internet as well as its significant impact on education, commerce, and societies throughout the world. Multiple conference tracks will address critical issues ranging from network engineering to user needs, from regulatory issues to the Internet's role as a conduit for social change, and from the transformation of education to the redefinition of commerce.

Topics

The Program Committee solicits abstracts of papers and suggestions for targeted sessions or panels which describe innovative developments, encourage vigorous discussion and further our understanding of the Internet's frontiers. The following list of topics is indicative of the scope of the conference and should not be interpreted as limiting submissions:

- High speed networks
- Advances in multimedia
- International infrastructure
- New hardware and software technologies
- Internationalization
- Security
- Mobile computing
- Collaborative work
- Information discovery
- Network Measurement and Management
- New industries and services
- Expanding Internet Access
- Electronic commerce
- Digital libraries and museums
- New paradigms for the user
- Advances in education
- Curriculum Innovations
- Legal issues
- Network learning
- Regulatory issues
- Social and cultural issues
- Regional issues
- The Internet as a universal communication medium

A limited amount of partial support may be available to assist presenters, generally from developing countries, to participate in INET '97.

Submission Guidelines

Register your interest in contributing to the INET '97 program by sending e-mail to `inet-program-interest@isoc.org`. You will receive complete details of the submission procedure and periodic updates from the Program Committee.

- The official language of the conference is English.
- Abstracts of papers and proposals for sessions or panels should be submitted in plain ASCII by *October 10, 1996* to: `inet-submissions@isoc.org`.
- Each abstract or proposal must contain a title or topic, the name(s) of the author(s), organizational affiliation(s), address(es), telephone and fax number(s), and e-mail address(es), and must identify a single point of contact if more than one author is listed.
- Each abstract should be between one and two pages in length and contain a list of key words or topics.
- Each panel or session proposal should indicate and justify the theme of the proposed session and include the names of suggested contributors.
- Selected submissions will be invited to contribute full papers. Final selection will be based on full papers.

Developing Countries Workshop

A *Network Training Workshop for Countries in the Early Stages of Internetworking* will be held in Kuala Lumpur during the week prior to INET '97. Further information will be available soon.

Tutorials

Full day tutorials focused upon a variety of special areas of interest in internetworking will be offered prior to the beginning of INET '97.

K–12 Workshop

A one day workshop for primary and secondary school teachers and administrators focusing on networking issues in the classroom will be held prior to the beginning of INET '97.

Internet Products and Services Showcase

An exhibition featuring Internet products and services will be held in conjunction with INET '97.

More information

More information can be obtained from:

INET '97
The Internet Society
12020 Sunrise Valley Drive, Suite 210
Reston, VA 20191–3429
USA

Telephone:	+1 703-648-9888
Fax:	+1 703-648-9887
E-mail:	`inet97@isoc.org`
For Program Information:	`inet-program-chair@isoc.org`

Up-to-the-minute information on INET '97 may be found at:

`http://www.isoc.org/inet97`

Interop Website Expands to Serve Community Year-round

If you've visited `www.interop.com` over the last few months, you may have noticed changes in the look and format of the website.

Interop Online is being revamped to serve the needs of the networking community year-round, complementing the shows and conferences that happen periodically around the world. In addition, the site uses technology to assist attendees in getting the most out of the events. Created and maintained by WebSource, a sister company to Interop under the Softbank Expos umbrella, the site is being run on the publishing business model. While Interop Online is accepting advertising and sponsorships, editorial content remains totally independent.

Features

The editorial sections will provide information about the latest networking technology and products, and draws heavily on the resources of the Interop conference. For example, the site currently includes a transcription of Dr. Peter Neumann's keynote talk on security from the April event in Las Vegas, as well as material from several of the most popular and highly-rated conference sessions. Interop instructors James Metzler, James Gaskin, and Ed Tittel, among others, will be providing regular columns, containing a mix of strategic advice and technical tips. The InteropNet section will be a major area, with technical feature stories detailing the latest experiments by the InteropNet team, and describing the technology used in the network. Other editorial features include book reviews, book excerpts from recent publications by Interop instructors, and a Resources page with dozens of links to technical guides, white papers, vendor sites, and other valuable places on the Internet. Interactive forums will allow visitors to the site to ask questions of the experts, and offer advice and tips of their own.

Buyer's Guide

The core of the site is the Buyer's Guide, with a large database containing company and product information on not only show exhibitors, but all companies in the networking industry. The user will be able to search products by name or category, and pull up product spec sheets, as well as product reviews from a variety of major publications in the computer and networking field.

Show Planner

Another major component of the site is the section devoted to information about upcoming shows and conferences, complete with course descriptions and instructor bios. A secure registration feature allows attendees to register online for the show, for the general conference, and for workshops and tutorials. The Show Planner allows the user to browse through the conference offerings and exhibitor map and create a Personal Schedule to help navigate the event. Once an attendee has created a personal login, he or she can come back and check their itinerary while at the show, or print out new show floor maps or conference schedules. Currently, the site is featuring complete information about NetWorld+Interop 96 Atlanta, coming up September 16–20. Interop Online is the place to get the latest updates on events at the show, such as the BOF schedule, and other sessions and events that are not scheduled until the last minute.

Feedback wanted

Interop Online is soliciting suggestions from *ConneXions* readers for other valuable features to include. In addition, we are looking for paid contributors to the site. If you would like to write articles or run a forum for Interop Online, please send a message to the editor at: `aryder@interop.com`.

Future NetWorld+Interop Dates and Locations

NetWorld+Interop 96	Atlanta, GA	September 16–20, 1996
NetWorld+Interop 96	Paris, France	October 8–11, 1996
NetWorld+Interop 96	London, England	Oct. 28–Nov. 1, 1996
NetWorld+Interop 96	Sydney, Australia	November 25–29, 1996
NetWorld+Interop 97	Singapore	April 7–11, 1997
NetWorld+Interop 97	Las Vegas, NV	May 5–9, 1997
NetWorld+Interop 97	Frankfurt, Germany	May 12–15, 1997
NetWorld+Interop 97	Tokyo, Japan	June 2–6, 1997
NetWorld+Interop 97	Atlanta, GA	October 6–10, 1997
NetWorld+Interop 97	Paris, France	October 20–23, 1997
NetWorld+Interop 97	London, England	October 27–30, 1997
NetWorld+Interop 97	Sydney, Australia	November 25–28, 1997

All dates are subject to change.

More information

Call 1-800-INTEROP or +1-415-578-6900 for more information. Or send e-mail to `info@interop.com` or fax to +1-415-525-0194. For the latest information about Interop DotCom and NetWorld+Interop as well as other SOFTBANK produced events, check our *Interop Online* home page at `http://www.interop.com`

NetWorld+Interop is produced by SOFTBANK Exposition and Conference Company, 303 Vintage Park Drive, Foster City, California 94404–1138, USA.

Write to *ConneXions*!

We'd love to hear your comments, suggestions and questions about anything you read in *ConneXions*. Our editorial address is given below. Use it for letters to the Editor, requests for the index of back issues, questions about particular articles etc.:

ConneXions—The Interoperability Report
303 Vintage Park Drive
Suite 201
Foster City
California 94404–1138
USA
Phone: +1 415-578-6900 or 1-800-INTEROP (Toll-free in the USA)
Fax: +1 415-525-0194
E-mail: `connexions@interop.com`
URL: `http://www.interop.com`

Subscription information

For questions about your subscription please call our customer service hotline: 1-800-575-5717 or +1 610-892-1959 outside the USA. This is the number for our subscription agency, Seybold Publications. Their fax number is +1 610-565-1858. The mailing address for subscription payments is: P.O. Box 976, Media, PA 19063–0976.

CONNEXIONS
303 Vintage Park Drive
Suite 201
Foster City, CA 94404-1138
Phone: 415-578-6900
FAX: 415-525-0194

ADDRESS CORRECTION
REQUESTED

FIRST CLASS MAIL
U.S. POSTAGE
PAID
SAN JOSE, CA
PERMIT NO. 1